Research Article

ASSOCIATION FOR PSYCHOLOGICAL SCIENCE

Psychological Science
2016, Vol. 27(9) 1215–1226
© The Author(s) 2016
Reprints and permissions:
sagepub.com/journalsPermissions.nav
DOI: 10.1177/0956797616654912
pss.sagepub.com
SAGE

Hidden Stages of Cognition Revealed in Patterns of Brain Activation

John R. Anderson, Aryn A. Pyke, and Jon M. Fincham
Carnegie Mellon University

Abstract

To advance cognitive theory, researchers must be able to parse the performance of a task into its significant mental stages. In this article, we describe a new method that uses functional MRI brain activation to identify when participants are engaged in different cognitive stages on individual trials. The method combines multivoxel pattern analysis to identify cognitive stages and hidden semi-Markov models to identify their durations. This method, applied to a problem-solving task, identified four distinct stages: encoding, planning, solving, and responding. We examined whether these stages corresponded to their ascribed functions by testing whether they are affected by appropriate factors. Planning-stage duration increased as the method for solving the problem became less obvious, whereas solving-stage duration increased as the number of calculations to produce the answer increased. Responding-stage duration increased with the difficulty of the motor actions required to produce the answer.

Keywords

cognitive neuroscience, computer simulation, neuroimaging, problem solving, reaction time

Received 11/28/15; Revision accepted 5/25/16

One of the striking features of the human mind is its ability to engage in complex intellectual operations. Analyzing the sequential structure of such activities has been a challenge for all methods, from behavioral to neural imaging approaches. There is a rich history in cognitive psychology of trying to infer the stages of a task by using effects of manipulations on total time and their interactions, such as in the additive-factors logic employed by Sternberg (1969). While such endeavors have had some success, they have been limited by the impoverished nature of behavioral data. The advent of brain imaging has provided much richer data that promises to help researchers better understand cognition. Nonetheless, it has been argued that aside from localization, methods such as functional MRI (fMRI) offer little additional insight into cognitive processes that could not have been obtained from behavioral experiments (e.g., Coltheart, 2013). Given the success of multivoxel pattern analysis (MVPA) at identifying the structure of concepts (Just, Cherkassky, Aryal, & Mitchell, 2010; Norman, Polyn, Detre, & Haxby, 2006; Pereira, Mitchell, & Botvinick, 2009), we investigated whether it could be used to track the sequence of mental stages in a complex task. We developed a method that can do just this—penetrate into the sequential structure of complex cognition.

MVPA can be used to recognize the distinct brain patterns that occur in different mental stages, but to do so, one must deal with the highly variable duration of these stages. In the present research, we considered a problem-solving task in which participants took anywhere from a couple of seconds to 30 s to solve problems. If we knew when participants were in different stages, we could train a pattern recognizer to identify the brain patterns associated with the stages, but we did not know a priori the temporal boundaries of the stages. If we knew what the brain patterns were that characterized each stage, we could use these patterns to estimate when participants were in different stages, but we did not know a priori what these patterns would be. The situation we faced was even more uncertain because we were not even sure how many cognitive stages were involved. Using this new method, we simultaneously discovered the stages, their associated brain patterns, and their durations on individual trials.

Corresponding Author:
John R. Anderson, Department of Psychology, Carnegie Mellon University, Pittsburgh, PA 15213
E-mail: ja@cmu.edu

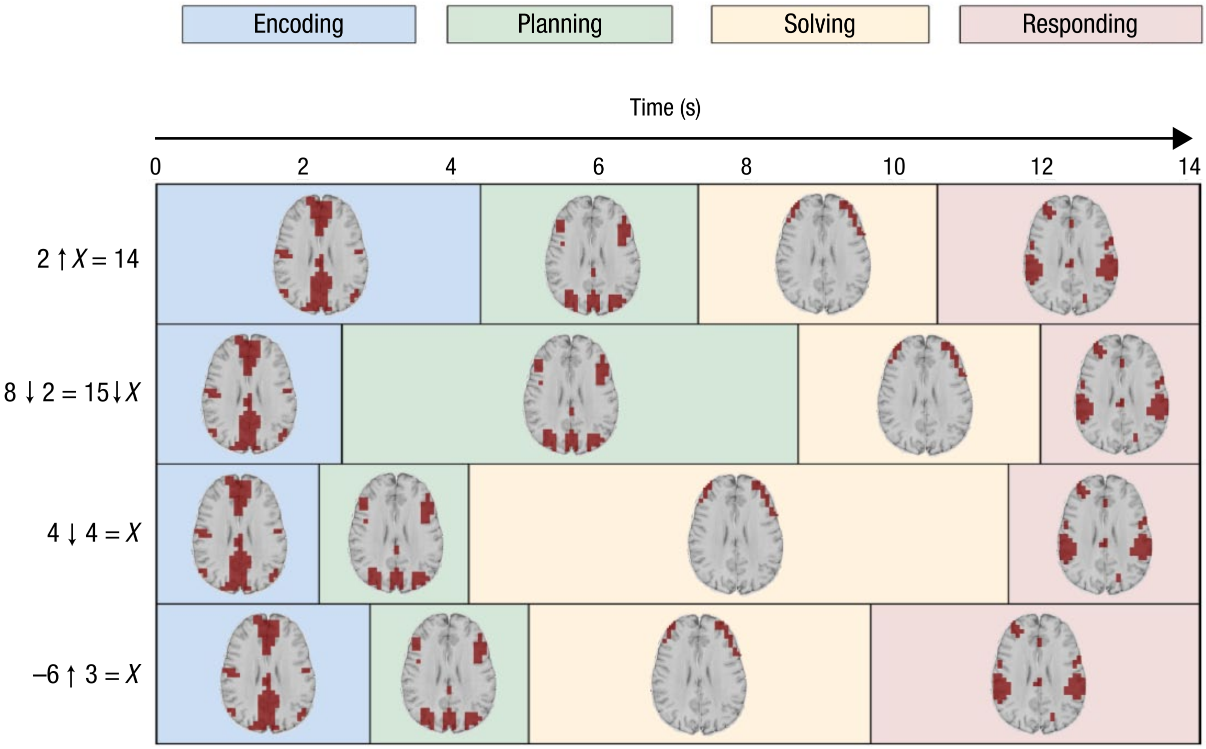

Fig. 1. Illustration showing the durations of the four stages associated with problem solving. In the four example problems, the arrows denote new mathematical operators that participants had learned. In each stage, the axial slice (x = 0 mm, y = 0 mm, z = 28 mm in Talairach space) highlights brain regions in which activation in that stage was significantly greater than the average activation during problem solving. Brain images are displayed with the left hemisphere on the right-hand side.

Figure 1 shows the output of this method, a parse of 4 trials (out of over 4,000) illustrating four stages in solving novel mathematical problems. The color coding reflects the durations of these stages. These particular problems in Figure 1 were chosen because each took seven 2-s scans (14 s) to complete, but the solvers spent different durations in the four stages. For purposes of illustration, the figure shows brain regions in a particular slice that tended to be most active in each stage.

The method that found the stages in Figure 1 uses a novel combination of hidden semi-Markov models (HSMMs; Yu, 2010) and MVPA to determine on a trial-by-trial basis how to associate fMRI scans with mental stages. Essentially, the HSMMs complement the MVPA by performing temporal pattern recognition. The HSMM assumes that the brain activity is the result of a participant going through a strict sequence of stages (i.e., one stage must complete before the next begins). It estimates a set of parameters that maximize the probability of the brain activity given a certain number of stages. The parameters include a specification of the mean brain-wide activity in each stage, called its *brain signature*. The brain signature is estimated by an MVPA of the brain activity in each cycle of an iterative estimation process. When the estimation process settles on a set of parameters, it parses each trial into the scan-by-scan probabilities that the participant is in a particular stage. These *stage-occupancy* profiles provide estimates of stage durations for a trial. The estimated stage durations for a trial are calculated as a sum of the stage probabilities over the scans that define that trial multiplied by the length of the scans. These trial-by-trial estimates allow us to determine which experimental factors affect the length of which stages.

Earlier work (Anderson, Betts, Ferris, & Fincham, 2010, 2012; Anderson, Fincham, Yang, & Schneider, 2012) showed that this HSMM-MVPA method could accurately recover the stages in tasks in which there were external markers of the stage boundaries (e.g., they coincided with overt actions or stimulus events at known time points). More recently (Anderson & Fincham, 2014; Anderson, Lee, & Fincham, 2014), we went a step further and applied these methods to identify the stages in tasks that did not afford such external markers. Without independent information about stage boundaries against which to train our system, we identified the stage structure that minimized the unpredicted variability in the imaging data. We went into this effort without strong expectations about the stage structure that would emerge.

Rather, these methods allowed us to discover the stages. While this was promising, we wanted to be sure that the stages were meaningful rather than artifacts of the statistical estimation. Therefore, in the current research, we tested predictions about these stages to establish their psychological reality.

The Current Research

The goal of the current research was to verify the psychological reality of the stages that are revealed by these HSMM-MVPA methods by testing whether experimental manipulations affected appropriate stages. Participants were taught two new mathematical operators, designated by ↓ and ↑, that mapped pairs of numbers onto values. Down-arrow problems, denoted $b \downarrow n = X$, could be solved by adding a sequence of n decreasing terms starting with b (e.g., $8 \downarrow 3 = 8 + 7 + 6 = 21$). Up-arrow problems, denoted $b \uparrow n = X$, could be solved by adding a sequence of n increasing terms starting with b (e.g., $8 \uparrow 3 = 8 + 9 + 10 = 27$). However, there were also formulas for solving these problems, and half the participants were not taught the addition rule but rather taught to solve the problems using the following equations:

$$b \downarrow n = b \times |n| - (n/2)(|n| - 1)$$

$$b \uparrow n = b \times |n| + (n/2)(|n| - 1)$$

These algebraic expressions might seem rather intimidating on first sight, but participants learned to solve problems using these equations with slightly greater facility than the participants who performed repetitive addition.

The down-arrow problems were similar to the problems used in our previous research (Anderson & Fincham, 2014), in which we discovered four stages in the solution of these problems—the encoding, planning, solving, and responding stages illustrated in Figure 1. We expected to find the same stages in the solution of the problems in this experiment. The goal of the current study was to test whether experimental manipulations would have the predicted effects on the durations of the different stages. This is like the additive-factors logic of Sternberg (1969), but we were using the activation patterns to measure the duration of each stage rather than inferring it from patterns of total latency. This study focused on two manipulations involving the two middle stages, planning and solving. The manipulations tested whether the planning-stage duration varies with the difficulty of coming up with a solution plan and whether the solving-stage duration varies with the difficulty of executing that plan.

The first manipulation involved the type of problem. The day before the scanner trials, participants learned their assigned method for solving problems and practiced calculating the value of up- and down-arrow problems with positive single-digit operands. On the second day, when they were in the scanner, they solved three types of problems (see Table 1). *Regular* problems were

Table 1. Description and Examples of the Problem Types Used in the Experiment

Problem type	Example problem	Comment
Regular (16 items)	$3 \downarrow 2 = X$ $3 \uparrow 2 = X$	Half ↑ problems, and half ↓ problems
Computational transfer		
Unknown b (8 items)	$X \downarrow 2 = 5$ $X \uparrow 2 = 7$	Participants may guess the value of X and then check it by computing the answer in the same way
Unknown n (8 items)	$3 \downarrow X = 5$ $3 \uparrow X = 7$	Participants may guess the value of X and then check it by computing the answer in the same way
Negative b (8 items)	$-3 \uparrow 2 = X$ $-2 \downarrow X = 6$	Algorithms apply as for regular problems
Negative n (8 items)	$4 \uparrow -2 = X$ $2 \downarrow -5 = X$	Equation accommodates negative n; negative reverses the direction of addition
Relational transfer		
Relating up and down (8 items)	$31 \uparrow 4 = 31 \downarrow X$ $19 \downarrow 4 = X \uparrow 4$	Solution process for the first problem is $b \uparrow N = b \downarrow (-N)$; solution process for the second problem is $b \downarrow N = (b - N + 1) \uparrow N$
Consecutive operand (8 items)	$35 \downarrow 3 = (34 \downarrow 2) + X$ $26 \downarrow 15 = (X \downarrow 14) + 26$	Solution process for the first problem is $b \downarrow N = (b - 1) \downarrow (N - 1) + b$; solution process for the second problem is $b \downarrow N = (b - 1) \downarrow (N - 1) + b$
Mirror problems (8 items)	$30 \downarrow 61 = X$ $5 \downarrow 5 = X \downarrow 6$	Solution process for the first problem is $b \downarrow (2b + 1) = 0$; solution process for the second problem is $b \downarrow b = b \downarrow (b + 1)$
Rule problems (8 items)	$5 \downarrow X = 7 \downarrow X$ $5 \uparrow 2 = 11 \uparrow X$	For the first problem, given any integer b, $b \uparrow 0 = 0$ and $b \downarrow 0 = 0$; for the second problem, given any integer b, $b \uparrow 1 = b$ and $b \downarrow 1 = b$

just like the ones they had been solving on the first day and served as a reference point for two types of transfer problems. The first kind of transfer problems were *computational* problems that required solving for different terms (e.g., $2 \uparrow X = 5$; answer: $X = 2$) or working with negative operands (e.g., $-2 \downarrow 5 = X$; answer: $X = -20$) but which could still be solved by repetitive addition or equation manipulation. The second kind of transfer problems were *relational* problems (e.g., $31 \downarrow 4 = 31 \uparrow X$; answer: $X = -4$) that could not be easily solved by the standard procedure but were fairly easy to solve if the participant came to an insight about the solution procedure. For instance, the solution to the relational problem $5 \downarrow X = 7 \downarrow X$ is $X = 0$, which is apparent if one realizes that $X = 0$ means no additions for the addition algorithm and multiplication by zero for the equation algorithm. Thus, relational problems should be harder to plan for but easier to solve, and therefore we predicted an interaction between the type of transfer problem and the duration of the planning versus solving intervals.

The other manipulation involved the second operand. We varied the second operands for regular and computational problems (but not relational problems) from 2 through 5. This value should strongly affect the duration of the solving interval for participants using the addition algorithm because it determined the number of additions. Thus, we predicted a three-way interaction among the size of the operand, the type of algorithm, and the stage (planning vs. solving): There should be a strong linear relation between operand size and duration only for the solving stage of the addition algorithm. Note that both of these predictions are more precise than simply predicting variation in total time; rather, they tested the ability of the HSMM-MVPA method to localize these effects in particular stages.

Method

Participants

The data for the analyses came from 80 participants (44% female, 56% male; mean age = 23.6 years, *SD* = 4.9), 40 of whom used the addition algorithm and 40 of whom used the equation algorithm.[1] Eighty participants were judged to be necessary for application of the HSMM-MVPA method.

Procedure

In a separate session prior to the scanner trials, participants learned about arrow operators by studying two worked-out examples ($4 \uparrow 3$ and $4 \downarrow 3$). Half the participants were exposed to equation-algorithm examples and half to addition-algorithm examples. Equation-algorithm examples illustrated the underlying computation, for instance, $4 \uparrow 3 = (4 \times |3|) + (|3|/2 \times [3 - 1]) = 12 + 3 = 15$, and $4 \downarrow 3 = (4 \times |3|) - (|3|/2 \times [3 - 1]) = 12 - 3 = 9$. Addition-algorithm examples revealed the underlying computation of that algorithm, for instance, $4 \uparrow 3 = 4 + 5 + 6 = 15$ and $4 \downarrow 3 = 4 + 3 + 2 = 9$. Participants in each group then practiced solving 90 such regular problems (45 per operation) of the form $b \uparrow n = X$ and $b \downarrow n = X$, where participants solved for X, and b and n were integers from 1 to 9 and 2 to 6, respectively. Problems were partitioned into eight blocks each with about equal numbers of up- and down-arrow problems. In the first and last blocks (12 problems each), participants were allowed to enter an expression before computing the final answer (e.g., addition: $4 \uparrow 3 = 4 + 5 + 6$; equation: $4 \uparrow 3 = 4 \times |3| + |3|/2 \times [3 - 2]$). The other six blocks (11 problems each), required all interpretation and computation to be done mentally.

During scanner trials, problems were presented one at a time on the screen in black font on a white background. There were eight blocks, and the first problem in each block was a warm-up regular problem (not analyzed), followed by a random ordering of two more regular problems, four computational problems, and four relational problems. The second operands in the regular problems and computational problems varied from 2 to 5. Over the eight blocks, each type of regular and computational problem was used with an equal number of the four possible second operands. Participants pressed the enter key when they had solved the problem and typed their answer using a keypad. A problem would time out if participants took longer than 30 s to compute an answer or longer than 5 s to type it. Otherwise, participants' answers appeared in blue font below the problem. After participants entered their response, their answer turned green if correct and red if incorrect, and the X in the problem was replaced with the correct answer, framed in a box. They had 7 s to study this feedback. To distract participants from thinking about the prior problem and allow brain activity to return to a relatively constant level, we inserted a repetition-detection task (12 s) after the feedback on each trial: A fixation cross (3 s) was followed by a series of letters (each for 1.25 s), and participants were instructed to press the enter key when the same letter appeared twice in a row.

Imaging acquisition

All imaging data were acquired at the Carnegie Mellon Scientific Imaging & Brain Research Center using a Siemens Verio 3 Tesla whole-body scanner equipped with a 32-channel radio-frequency head coil. There were eight functional scanning runs. Functional T2*-weighted images were acquired using a gradient-echo echo-planar imaging (EPI) acquisition sequence with the following parameters:

3.2-mm slice thickness, 34 oblique-axial slices, 2,000-ms repetition time (TR), 30-ms echo time (TE), 79° flip angle, 64 × 64 matrix, and 200-mm field of view (FOV). This produces voxels that are 3.2 mm thick and 3.125 × 3.125 mm^2 in plane. The anterior commissure-posterior commissure line was centered on the 11th slice from the bottom scan slice, which yielded nearly full brain coverage with the exception of the most inferior portions of the temporal lobes, brain stem, and cerebellum. High-resolution structural T2-weighted images were also acquired with the following parameters: 3.2-mm slice thickness, 34 oblique-axial slices, 3,700-ms TR, 84-ms TE, 120° flip angle, 640 × 640 matrix, and 200-mm FOV.

Imaging analysis

Functional data were analyzed using a combination of custom code and the software package AFNI (Cox, 1996; Cox & Hyde, 1997). Prior to statistical analysis, the first four EPI volumes from each run were discarded to minimize magnetization equilibration effects. Preprocessing steps consisted of motion correction using six-parameter rigid-body registration to align each EPI volume to the first EPI volume in the run, coregistration of the T2 structural volume to the first EPI volume in the run, slice-time centering at 1 s, and normalization such that the voxel time series within runs had a mean value of 100. T2 structural volumes were normalized to a common reference structural MRI using a 12-parameter 3-D affine registration. The functional data were then spatially normalized into this space by applying the computed transformation, preserving voxel dimensionality, and spatially smoothing with a 6-mm full-width-at-half-maximum 3-D Gaussian filter to accommodate individual differences in anatomy.

As a step of dimension reduction, the within-brain voxels in each slice were aggregated into 12,481 larger 2 × 2 voxel regions (by 3.2 mm for the slice). To ensure comparable data quality across all participants, we censored those regions that showed an excess of extreme values for any particular participant.[2] The censored regions likely reflected differences in anatomy as most were located mainly on the topmost and bottommost slices, as well as around the perimeter of the brain. This process yielded a total of 8,755 high-data-quality regions.[3]

We applied a deconvolution step to the functional data from each of the 8,755 regions. We assumed the blood-oxygen-level-dependent (BOLD) response was produced by the convolution of an underlying activity signal with a hemodynamic response function. Assuming the hemodynamic response function of the Statistical Parametric Mapping difference of gammas (Friston et al., 1998: g = gamma(6, 1) – gamma(16, 1)/6), we used a Wiener filter (Glover, 1999) with a noise parameter of .1 to deconvolve the BOLD response into an inferred activity signal for each region. This results in approximately shifting the BOLD signal backward in time by two scans (4 s) to compensate for its lag relative to neural activity. The inferred signal averages for each region were then z-scored to eliminate any mean activity differences and to focus on sequential structure. This also guarantees that each region contributes equal variance to the analysis.

To deal with the highly correlated nature of brain signals in different regions, we performed a spatial principal component analysis (PCA) on the data and used the scores of the first 20 principal components (see the Supplemental Material available online for the 20 principal components). Churchill, Yourganov, and Strother (2014) reported that using a PCA as a regularization method maximizes reproducibility and global signal-to-noise ratio. To ensure that all dimensions were equally weighted and all participants equally represented in the HSMM-MVPA, we z-scored each dimension for all scans during which problems were correctly solved for each participant.

The resulting 20 principal component dimensions for each trial, from stimulus onset to response completion, were the input the HSMM-MVPA process. The MVPA estimates brain patterns by assuming that each trial yields a sample from a 20-dimensional normal distribution around a set of mean principal component values for that stage. These mean values define the brain signature for that stage. Whole-brain activation patterns for a stage can be reconstructed from the 20 principal component means that define its brain signature. A traditional Markov model would assume that stage durations do not vary; however, the semi-Markov model we used assumed that, on different trials, participants spend different amounts of time in a stage. Variability in each stage's duration is represented by a gamma distribution.

There are many possible ways to partition the scans of a particular trial into different stages.[4] The HSMM can efficiently compute the summed probability of all such partitions given the means estimated for the brain signatures and the gamma distributions. The stage occupancies in Figure 2 are the summed probabilities over all such partitions that a scan is in a stage. The expectation maximization associated with HSMMs (Yu, 2010) estimates the brain signatures and gamma distributions for each stage to maximize the probability of the data on all trials.

Results

Figure 2 displays the basic behavioral results: accuracy and response latency broken down by problem type and algorithm. Of the errors, 22.6% were due to time-outs, and the rest were wrong answers. Accuracy was strongly affected by both factors of interest—problem type, $F(2, 156) = 246.44$, $p < .0001$, and algorithm, $F(1, 78) = 15.80$,

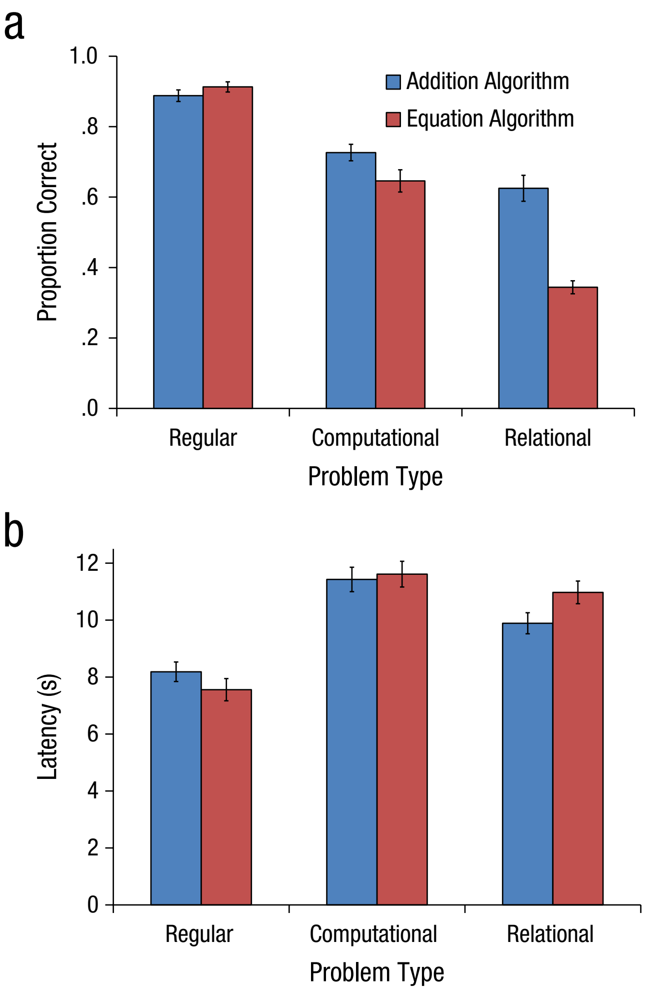

Fig. 2. Mean accuracy (a) and response latency (b) for correctly solved problems as a function of problem type and algorithm used to solve the problems. Error bars show ±1 *SEM*.

$p < .0005$. While participants were overall more accurate using the addition than the equation algorithm, there was an interaction between algorithm and problem type, $F(2, 156) = 35.86$, $p < .0001$, such that this advantage was strongest for relational problems. Equation participants solved regular problems with slightly more accuracy than addition participants, but this was not significant, $t(78) = -1.36$, $p > .10$. Addition participants solved computational problems with slightly more accuracy than equation participants, $t(78) = 2.08$, $p < .05$, and showed a large advantage for relational problems, $t(78) = 6.91$, $p < .0001$.

One participant (in the equation group) failed to correctly solve any relational problems. Figure 2b gives the mean latency for correct responses from the remaining participants. There was no significant main effect on the overall mean times of the two algorithm groups, $F(1, 77) = 0.16$. There was a highly significant effect of problem type, $F(2, 154) = 126.94$, $p < .0001$, and a significant interaction between problem type and algorithm, $F(2, 154) = 6.72$, $p < .005$. The latency interaction mirrored the pattern for accuracy: Addition participants, compared with equation participants, were slower on regular problems (8.19 s vs. 7.56 s) but faster on computational problems (11.43 s vs. 11.61 s) and relational problems (9.89 s vs. 10.97 s). The effect for relational problems was significant, $t(77) = 2.00$, $p < .05$, while the other pairwise comparisons were not significant.

While analyses of total latency did not address the stage predictions (see the introduction), these results did establish clear effects of both problem type and algorithm. It remained to be determined whether these factors affected the planning and solving stages as predicted. But first we looked for evidence that these four stages existed.

Evidence for four stages

To verify that participants went through four cognitive stages in solving these problems, we fitted models with different numbers of stages to the data of half of the participants (20 from each algorithm) and then assessed how well these parameters fitted the data of the other 40 participants. A more complex model (with more stages) would fit the data of the trained half of the participants better, but if it was overfitting, it would not predict the data of the tested half of the participants better.

Figure 3 shows how the mean log likelihood of the predicted participant data varied with the number of assumed stages. Starting from a single-stage model, prediction accuracy increased substantially when additional stages were added, up until five stages. We measured the significance of the improvement using a sign test to determine how many participants' data were better fitted with more stages. Up until five stages, each additional

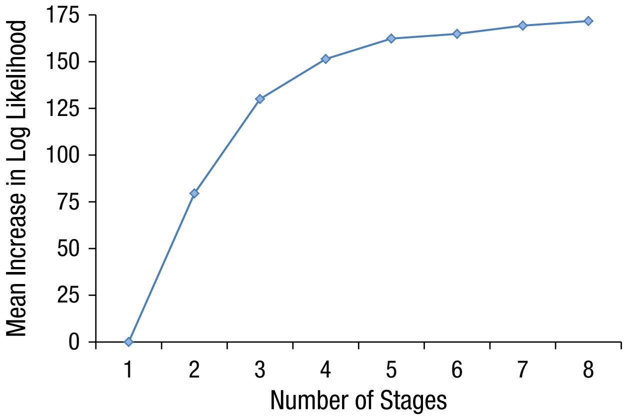

Fig. 3. Mean increase in log likelihood for a one-stage model relative to a model with two through eight stages. The mean log likelihood of the one-stage model was −2,747.9. Data from 40 participants were used to calculate the log likelihood of the model fitting the other 40 participants.

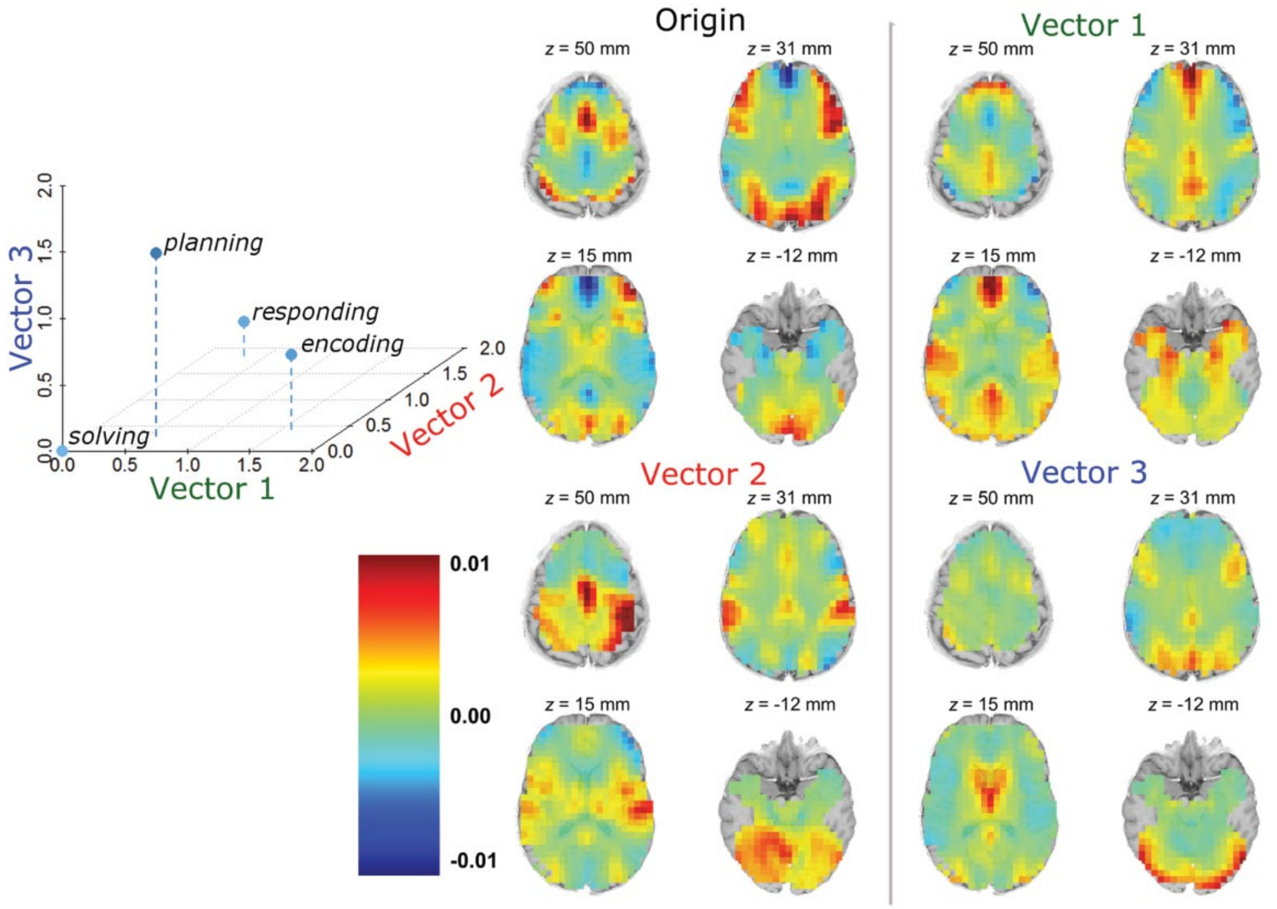

Fig. 4. The four brain signatures placed in a 3-D space where the activity of a stage is a sum of the activity of the signature in the solving stage plus a sum of the three vectors weighted by their coordinates in the space. The heat maps illustrate the proportion of change in activation relative to baseline. The coordinates of the stages are as follows (in Talairach space)—encoding: $x = 1.61$, $y = 0.37$, $z = 0.58$; planning: $x = 0.58$, $y = 0.28$, $z = 1.38$; solving: $x = 0$, $y = 0$, $z = 0$; and responding: $x = 0.37$, $y = 1.78$, $z = 0.28$. Brain images are displayed with the left hemisphere on the right-hand side.

stage resulted in improvement to the fit of most participants' data. The least significant case was between four and five stages, where 34 of the 40 participants' data were predicted better with five stages ($p < .0001$), while only 25 of the participants' data were fitted better by six stages ($p > .1$). Examination of the principal component values of the five-stage model (see the Supplemental Material) shows that it had the same first three stages as the four-stage model. The fourth and fifth stages were highly correlated with each other and with the last stage of the four-stage model, but the fifth stage had more extreme values. Because the last two stages did not represent distinct patterns (just differences in magnitude), we analyzed the four-stage solution in which each stage had a very distinct pattern in terms of the 20 principal component values that described its brain signature.

The brain signatures that defined each stage were complex multidimensional patterns (brain signatures are given in the Supplemental Material), but as there were only four points in that complex space, they spanned a 3-D subspace. This 3-D subspace can be used to provide a useful differentiation of the stage structures. Figure 4 shows the position of these four stages in that 3-D space, setting the origin to be the brain signature for the solving stage and selecting axes to best separate the stages. The activity for the other stages was constructed by adding to the solving-stage pattern the vectors associated with the three dimensions weighted by the coordinates of the stages. This representation reflected the fact that the solving stage had the lowest average activity. Each of the other stages can be seen as adding something to the solving-stage pattern. The other stages had a large value on one dimension and small values on the other dimensions: The encoding stage had a large value on Vector 1 that had increased activity in default mode regions, such as posterior cingulate and polar frontal areas (Buckner, Andrews-Hanna, & Schacter, 2008). We consider this pattern to reflect external orientation (Mayer, Dorflinger, Rao, & Seidenberg, 2004). The planning stage had a large value on Vector 2, which has increased activity along the ventral visual stream and parietal, prefrontal, and caudate regions. The responding stage had a large value on Vector 3, which showed particularly high activity in the left motor area.

In addition to the parameters that define the brain signatures of the stages, we estimated parameters to characterize the distribution of stage durations across trials. Figure 5a illustrates the gamma distributions estimated for the four stages. The solving stage had the most striking distribution, with many very short durations and a very long tail; 10.7% of all solving-stage trials had durations less than 50 ms.[5] These distributions of stage durations reflect differences among problem types as well as other sources of variability. Two-thirds of these very brief solving stage durations involved either relational problems or regular and computational problems with second operands of 2, which arguably involve very little arithmetic. Although we estimated continuous distributions, they were being estimated from data that came in 2-s scans. Figure 5b illustrates how well the convolution of the estimated stage durations in Figure 5a matches the distribution of the observed number of scans per problem (r = .975).

Effects of differences among conditions

The duration of trials for participants who used the addition and equation algorithms varied depending on the problem type. Activation patterns may also vary, and these effects may be localized in one or more stages. As discussed earlier, the HSMM-MVPA method generated estimates of how much time each participant spent on each stage of each trial. We used these stage occupancies (trial-by-trial information about when stages occurred) to identify where latency and activation effects were located and test their consistency over participants. We fitted a single model to all participants and conditions and used its stage occupancies for identifying such effects. Using a single model provides a constant measuring instrument for these effects.

A number of left-lateralized regions were more active throughout the trials for participants executing the addition algorithm: middle temporal gyrus, superior temporal gyrus, and angular gyrus. These regions all showed the same interaction with stage: The difference between addition and equation participants was greatest during the first encoding stage and least during the response stage. Grabner et al. (2007) have reported that these same regions were more active in more competent participants during a mathematical problem-solving task. This is consistent with the superior performance of the addition participants in our experiment. The interaction suggests that engagement of these regions is most important early in the problems. (The Supplemental Material provides a detailed report of the effects of condition on brain activation.)

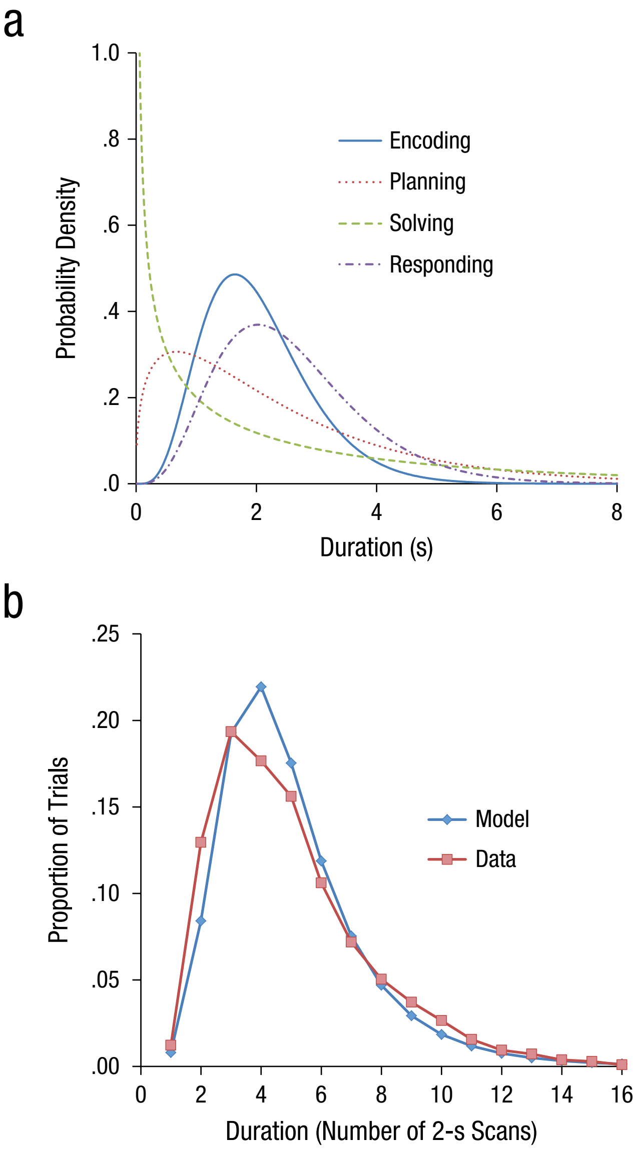

Fig. 5. Results of the trial-duration analyses. The graph in (a) shows the probability density function for each of the four stages. The graph in (b) shows the distribution of trial durations in the data and the distribution predicted by convolving the durations of the four stages.

The primary focus of this research was predictions regarding the latency patterns of the stages. Table 2 reports the results of two-way analyses of variance on the four stage durations (on trials with correct responses) with the factors algorithm (addition, equation) and problem type (regular, computational, relational). Because 1 participant in the equation condition did not correctly solve any relational problems, there were only 79 participants in this analysis. There were no effects of these factors predicted for the encoding and responding stages.

Table 2. Results From the Analysis of Variance on Stage Duration

		Stage			
Effect	Degrees of freedom	Encoding	Planning	Solving	Responding
Algorithm	1, 77	0.09	0.68	0.02	0.76
Problem type	2, 154	0.41	143.68**	37.32**	8.92*
Algorithm × Problem Type	2, 154	0.61	0.11	24.68**	9.87**

*$p < .001$. **$p < .0001$.

There were no significant effects for the encoding stage. There were some effects on responding time, which we will consider after examining the effects on planning and solving time that were the focus of our manipulations and predictions. Planning and solving times both showed a strong effect of problem type. Solving time also interacted strongly with algorithm.

Effect of problem type. Figure 6 shows the duration of the four stages for the three problem types and two algorithms. As indicated in the analysis of variance, there was no effect of problem type, algorithm, or their interaction on the encoding stage. While there was a significant effect of these factors on the responding stage, it was tiny compared with their effects on the planning and solving stages. Not surprisingly, the practiced regular problems were solved the fastest during the planning and solving stages. As predicted, computational problems were solved faster than relational problems in the planning stage, $t(78) = 2.66$, $p < .01$, but much slower in the solving stage, $t(78) = 5.16$, $p < .0001$.

Effect of second operand. Figure 7 displays the effect of varying the second operand, averaged over regular and computational problems. As predicted, there was a linear effect of second operand (n) on the solving stage only for participants who used the addition algorithm; this resulted in a highly significant three-way interaction among stage, second operand, and algorithm, $F(9, 684) = 44.28$, $p < .0001$. While it is not surprising that the size of the second operand affected overall solution time using the addition algorithm, it is striking that this HSMM-MVPA analysis method so clearly localized the effect in the solving stage.

Effect of keying difficulty. The effect of problem type on the duration of the responding stage (see Table 2) reflects, at least in part, differences in the difficulty of keying responses, which was not equated across problem types. Answers to regular problems averaged 1.76 keystrokes, computational problems averaged 1.60 keystrokes, and relational problems averaged 1.65 keystrokes. Negative answers were given only for computational problems (24.3%) and relational problems (15.1%). The negative

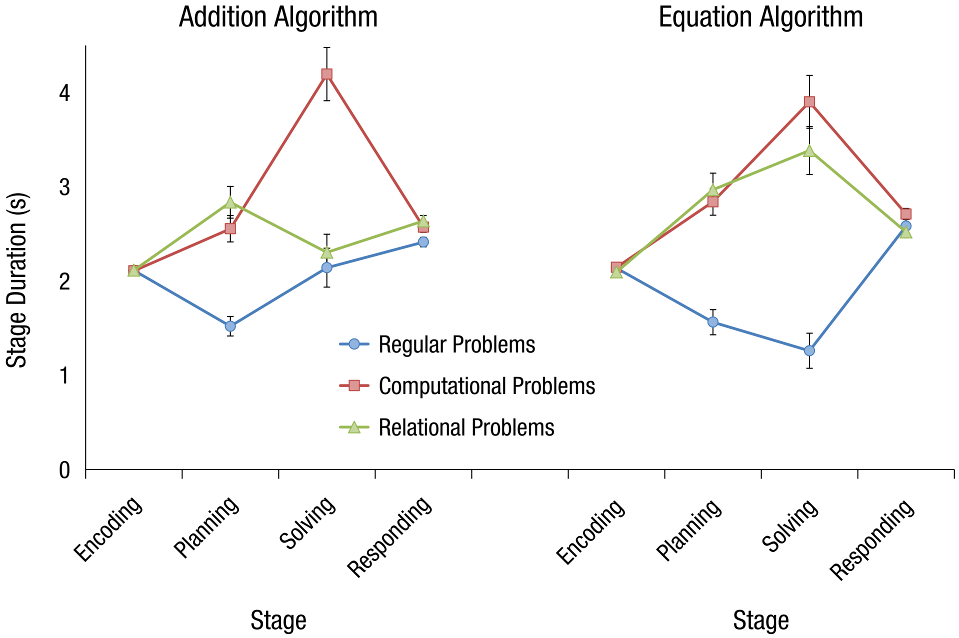

Fig. 6. Mean duration of each of the four stages as a function of problem type, plotted separately for participants who used the addition and equation algorithms. Error bars show ±1 *SEM*.

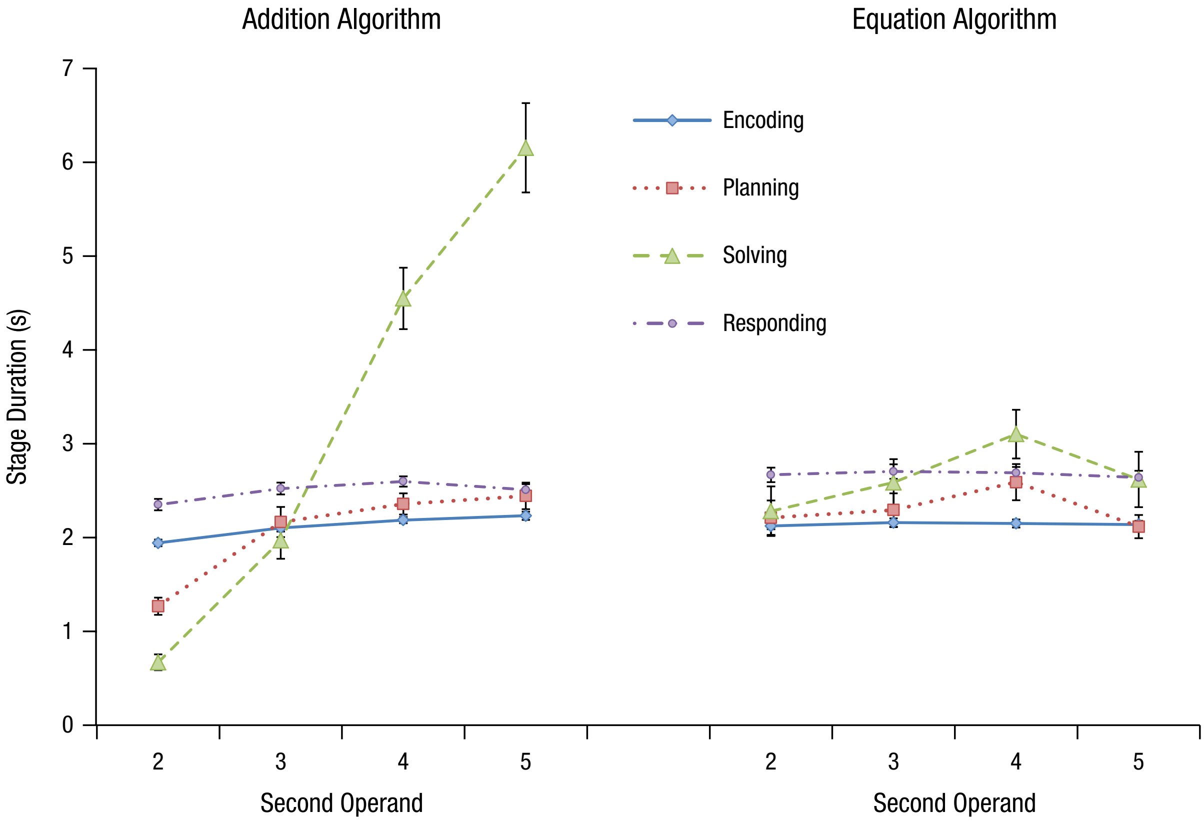

Fig. 7. Mean duration of each of the four stages as a function of the second operand, plotted separately for participants who used the addition and equation algorithms. Results are averaged across participants who solved regular and computational problems. Error bars show ±1 *SEM*.

key, located near the upper right edge of the keypad, was somewhat awkward to reach, particularly for small hands. For example, in Figure 1, the last problem is the only one with a negative answer (−15) and is characterized by a noticeably longer responding time.

To sort out the effects of problem type, number of keystrokes in the answer, and use of the negative key, we performed a mixed-effects analysis of responding times. Including number of keystrokes resulted in a decisive 853 decrease in Bayesian information criterion (BIC), including the existence of a negative answer resulted in a 790 BIC decrease, and including both resulted in a 1,014 BIC decrease. In contrast, adding problem type to the model did not decrease the BIC score but rather increased it by 47. Thus, it would appear that the only effects in the responding stage involve keying difficulty. Each key pressed resulted in a 0.34-s increase in responding time, and if the negative key had to be pressed, there was an additional 0.49-s increase. Table 3 gives the average duration of the responding stage for different combinations of one- and two-digit answers with and without a negative sign.

Table 3. Mean Duration of the Responding Stage for One- and Two-Digit Answers With and Without a Negative Sign

	Problem type		
Type of answer	Regular	Computational	Relational
Positive answer			
One digit	2.38 s	2.32 s	2.19 s
Two digit	2.55 s	2.61 s	2.69 s
Negative answer			
One digit	—	3.23 s	3.00 s
Two digit	—	3.40 s	3.37 s

Discussion

Problem-solving time is highly variable in duration, with large variation among participants and among problems, and even large variations within the same individual on similar problems. Nonetheless, the HSMM-MVPA method is able to identify a sequence of stages characterized by distinct brain patterns. Reflecting the limitations of the temporal resolution of fMRI, these stages tend to occupy many seconds, and we cannot penetrate into the detailed cognitive steps within each stage. Still, as Figure 1 illustrates, even for problems with the same overall response time, we can identify considerable variability in the duration of the different stages. The goal of this research was to establish that the variability in the duration of these stages was not just random, but was related to the characteristics of the problem being solved and the solution algorithm.

These analyses have provided strong evidence for the assumed identity of three of the stages. The manipulated factors had their predicted effects on the planning and solving stages. Although the experiment did not involve a balanced manipulation of keying difficulty, a mixed-effects analysis revealed that the factors determining the duration of the responding stage were indeed the number of keys and the difficulty of keying negative signs. The HSMM-MVPA method was able to localize effects of conditions to appropriate stages even though the analysis had no information about what condition-specific problems were in.

While this research focused on a mathematical problem-solving task, the results are encouraging for the general use of the HSMM-MVPA methods to parse complex cognition into distinct stages. The distinctive cognitive demands of each stage will produce a brain pattern that can be used to estimate temporal boundaries of that stage on each trial. One can then determine which various factors of interest affect the durations of these different cognitive stages. Researchers no longer need be left looking only at total latency.

Action Editor

John Jonides served as action editor for this article.

Author Contributions

J. R. Anderson performed the hidden semi-Markov models-multivoxel pattern analysis modeling. A. A. Pyke designed the experiment and analyzed the nonimaging data. J. M. Fincham analyzed the imaging data.

Acknowledgments

The data and code for the analyses reported in this article, along with the Supplemental Material, are available at http://act-r.psy.cmu.edu/?post_type=publications&p=19042.

Declaration of Conflicting Interests

The authors declared that they had no conflicts of interest with respect to their authorship or the publication of this article.

Funding

This work was supported by National Science Foundation Grant No. DRL-1420008 and by a James S. McDonnell Foundation Scholar Award.

Supplemental Material

Additional supporting information can be found at http://pss.sagepub.com/content/by/supplemental-data

Notes

1. We previously published an earlier nonstage analysis of half of these participants (Pyke, Betts, Fincham, & Anderson, 2015). Crossed with the factor of algorithm type (equation vs. addition) was whether participants had a spatial referent or not, but this spatial-referent factor had no impact on our results.

2. Regions were eliminated if (a) scans showed activity less than 50% of the mean signal value for a block, (b) more than 0.02% of scans had signals more than 25% from the mean, and (c) more than 0.2% of scans had signals more than 10% from the mean.

3. The Supplemental Material contains an analysis showing that the conclusions do not change if the analysis is based on the original voxels.

4. If there are n scans and m stages, the number of possible partitions into stages is $(n + m - 1)!/(n! \times (m - 1)!)$.

5. Estimated solving-stage durations were less than 50 ms on 11.5% of trials. In the encoding, responding, and planning stages, only 0.02%, 0.02%, and 2.7% of trials, respectively, had such brief durations.

References

Anderson, J. R., Betts, S., Ferris, J. L., & Fincham, J. M. (2010). Neural imaging to track mental states while using an intelligent tutoring system. *Proceedings of the National Academy of Science, USA*, *107*, 7018–7023.

Anderson, J. R., Betts, S. A., Ferris, J. L., & Fincham, J. M. (2012). Tracking children's mental stages while solving algebra equations. *Human Brain Mapping*, *33*, 2650–2665.

Anderson, J. R., & Fincham, J. M. (2014). Extending problem-solving procedures through reflection. *Cognitive Psychology*, *74*, 1–34.

Anderson, J. R., Fincham, J. M., Yang, J., & Schneider, D. W. (2012). Using brain imaging to track problem solving in a complex state space. *NeuroImage*, *60*, 633–643.

Anderson, J. R., Lee, H. S., & Fincham, J. M. (2014). Discovering the structure of mathematical problem solving. *NeuroImage*, *97*, 163–177.

Buckner, R. L., Andrews-Hanna, J. R., & Schacter, D. L. (2008). The brain's default network: Anatomy, function, and relevance to disease. *Annals of the New York Academy of Sciences*, *1124*, 1–38.

Churchill, N. W., Yourganov, G., & Strother, S. C. (2014). Comparing within-subject classification and regularization methods in fMRI for large and small sample sizes. *Human Brain Mapping*, *35*, 4499–4517.

Coltheart, M. (2013). How can functional neuroimaging inform cognitive theories? *Perspectives on Psychological Science*, *8*, 98–103.

Cox, R. W. (1996). AFNI: Software for analysis and visualization of functional magnetic resonance neuroimages. *Computers and Biomedical Research*, *29*, 162–173.

Cox, R. W., & Hyde, J. S. (1997). Software tools for analysis and visualization of fMRI data. *NMR in Biomedicine*, *10*, 171–178.

Friston, K. J., Fletcher, P., Josephs, O., Holmes, A. P., Rugg, M. D., & Turner, R. (1998). Event-related fMRI: Characterizing differential responses. *NeuroImage*, *7*, 30–40.

Glover, G. H. (1999). Deconvolution of impulse response in event-related BOLD fMRI. *NeuroImage*, *9*, 416–429.

Grabner, R. H., Ansari, D., Reishofer, G., Stern, E., Ebner, F., & Neuper, C. (2007). Individual differences in mathematical competence predict parietal brain activation during mental calculation. *NeuroImage*, *38*, 346–356.

Just, M. A., Cherkassky, V. L., Aryal, S., & Mitchell, T. M. (2010). A neurosemantic theory of concrete noun representation based on the underlying brain codes. *PLoS ONE*, *5*(1), Article e8622. doi:10.1371/journal.pone.0008622

Mayer, A. R., Dorflinger, J. M., Rao, S. M., & Seidenberg, M. (2004). Neural networks underlying endogenous and exogenous visual–spatial orienting. *NeuroImage*, *23*, 534–541.

Norman, K. A., Polyn, S. M., Detre, G. J., & Haxby, J. V. (2006). Beyond mind-reading: Multi-voxel pattern analysis of fMRI data. *Trends in Cognitive Sciences*, *10*, 424–430.

Pereira, F., Mitchell, T., & Botvinick, M. (2009). Machine learning classifiers and fMRI: A tutorial overview. *NeuroImage*, *45*, S199–S209.

Pyke, A., Betts, S., Fincham, J. M., & Anderson, J. R. (2015). Visuospatial referents facilitate the learning and transfer of mathematical operations: Extending the role of the angular gyrus. *Cognitive, Affective, & Behavioral Neuroscience*, *15*, 229–250.

Sternberg, S. (1969). The discovery of processing stages: Extensions of Donders' method. *Acta Psychologica*, *30*, 276–315.

Yu, S.-Z. (2010). Hidden semi-Markov models. *Artificial Intelligence*, *174*, 215–243.